

## **IN THE CLAIMS:**

Please substitute the following claims for the same numbered claims in the application.

1. (Currently Amended) A method of providing access for a plurality of application-level users to an application comprising a plurality of resource class components collectively executing on multiple networked machines, the method comprising steps of:

(i) receiving an incoming flow of requests from application-level users to use an application and components of said application;

(ii) providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or more machines, to service the incoming requests from respective application-level users to use the application;

(iii) directing each of the incoming requests to a particular application instance of an appropriate resource class component;

(iv) monitoring, for each of the application-level users, the number of request serviced by the application instances of the resource class components of the application; and

(v) increasing or decreasing the number of application instances of one or more resource class components in response to the monitored number of requests for each resource class component[[]];

maintaining a record of the current rate of requests received from respective application-level users, based on the monitored number of serviced requests; and

collectively and automatically allocating fractions of different resource class components to a particular application-level user in response to the increased or decreased number of application instances of one or more resource class components.

2. (Currently Amended) The method as claimed in claim 1, further comprising the step of: directing each of the incoming requests from respective application-level users to a particular application instance of an appropriate resource class component from a respective set of one or more application instances of each resource class component, said particular application instance being identified as the least loaded of the application instances of the appropriate resource class component from that respective set.

3. (Currently Amended) The method as claimed in claim 1, wherein the step of providing application instances of each resource class component further comprises the steps of:

initiating one or more application instance of one or more resource class on a plurality of machines to service incoming requests to use the application; and

terminating one or more application instances of each resource class on a plurality of machines to service incoming requests to use the application.

4. (Currently Amended) The method as claimed in claim 1, wherein requests from application-level users to use the application are stored in a queue for execution by a particular application instance of the appropriate resource class on a first-in-first-out basis.

5. (Currently Amended) The method as claimed in claim 1, further comprising the step of: maintaining a record of service obligations to respective application-level users.

6. (Currently Amended) The method as claimed in claim 5, further comprising the step of:

increasing or decreasing, for each of the application-level users, the number of application instances of each resource class component in response to the monitored number of requests for each resource class component, wherein the service obligations to respective application-level users are at least met.

7. (Cancelled).

8. (Currently Amended) The method as claimed in claim 7, wherein said step of increasing or decreasing the number of application instances of said one or more resource classes in (i) at least partly based upon said recorded current rate of requests received from respective application-level users, and (ii) at least partly based on predetermined information that correlates changes in request rates with changes in the corresponding number of application instances of said one or more resource classes required to service said request rates.

9. (Currently Amended) The method as claimed in claim 1, wherein one or more of the application-level users are organizations, and the requests are generated by individuals associated with the respective organization.

10-12. (Canceled)

13. (Currently Amended) A method of providing access for a plurality of application-level users to an application comprising a plurality of resource class components collectively executing on multiple networked machines, the method comprising steps of:

receiving an incoming flow of requests from application-level users to use an application and components of said application;

providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or more machines, to service the incoming requests from the application-level users to use the application;

monitoring, for each of the application-level users, the resources currently available and resources currently consumed by the requests serviced by application instances of the resource class components of the application; and

maintaining (i) a record of resources currently available to respective application-level users; and (ii) a record of resources currently consumed by respective application-level users; both records of said resources being maintained in respect of each of the one or more application instances of each resource class components[.]; and

adjusting the respective numbers of said one or more application instances of each resource class component; and

collectively and automatically allocating fractions of different resource class components to a particular application-level user in response to a fluctuating number of application instances of one or more resource class components.

wherein said application instances of each resource class component are adjusted for each application-level user based (i) at least partly on said records of resources currently available and currently consumed by respective application-level users, and (ii) at least partly on predetermined information that estimates the number of each resource class components required to service requests for said application instances of the resource class components.

14. (Cancelled).

15. (Currently Amended) A system for providing access for a plurality of application-level users to an application comprising a plurality of resource class components collectively executing on multiple networked machines, the system comprising:

(i) means for receiving an incoming flow of requests from application-level users to use an application and components of said application;

(ii) means for providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or more machines, to service the incoming requests from respective application-level users to use the application;

(iii) means for directing each of the incoming requests to a particular application instance of an appropriate resource class component;

(iv) means for monitoring, for each of the application-level users, the number of requests serviced by the application instances of the resource class components of the application; and

(v) means for increasing or decreasing the number of application instances of one or more resource class components in response to the monitored number of requests for each resource class component[.];

means for maintaining a record of the current rate of requests received from respective application-level users, based on the monitored number of serviced requests; and

means for collectively and automatically allocating fractions of different resource class

components to a particular application-level user in response to the increased or decreased number of application instances of one or more resource class components.

16. (Currently Amended) A computer software program, recorded on a medium and capable of execution by computing means able to interpret the computer software program, for providing access for a plurality of application-level users to an application comprising a plurality of resource class components collectively executing on multiple networked machines, the computer software program comprising:

(i) code means for receiving an incoming flow of requests from application-level users to use an application and components of said application;

(ii) code means for providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or more machines, to service the incoming requests from respective application-level users to use the application;

(iii) code means for directing each of the incoming requests to a particular application instance of an appropriate resource class component;

(iv) code means for monitoring, for each of the application-level users, the number of requests serviced by the application instances of the resource class components of the application; and

(v) code means for increasing or decreasing the number of application instances of one or more resource class components in response to the monitored number of requests for each resource class component[.];

code means for maintaining a record of the current rate of requests received from

respective application-level users, based on the monitored number of serviced requests; and  
code means for collectively and automatically allocating fractions of different resource  
class components to a particular application-level user in response to the increased or decreased  
number of application instances of one or more resource class components.

17-18. (Canceled)

19. (Currently Amended) A system for providing access for a plurality of application-level users to an application comprising a plurality of resource class components collectively executing on multiple networked machines, the system comprising:

means for receiving an incoming flow of requests from application-level users to use an application and components of said application;

means for providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or more machines, to service the incoming requests from the application-level users to use the application;

means for monitoring, for each of the application-level users, the resources currently available and resources currently consumed by the requests serviced by application instances of the resource class components of the application; ~~and~~

means for maintaining (i) a record of resources currently available to respective application-level users; and (ii) a record of resources currently consumed by respective application-level users; both records of said resources being maintained in respect of each of the one or more application instances of each resource class components[.];

means for adjusting the respective numbers of said one or more application instances of each resource class component; and

means for collectively and automatically allocating fractions of different resource class components to a particular application-level user in response to a fluctuating number of application instances of one or more resource class components,

wherein said application instances of each resource class component are adjusted for each application-level user based (i) at least partly on said records of resources currently available and currently consumed by respective application-level users, and (ii) at least partly on predetermined information that estimates the number of each resource class components required to service requests for said application instances of the resource class components.

20. (Currently Amended) A computer software program recorded on a medium and able to be executed by computing means able to interpret the computer software program, for providing access for a plurality of application-level users to an application comprising a plurality of resource class components collectively executing on multiple networked machines, the computer software program comprising:

code means for receiving an incoming flow of requests from application-level users to use an application and components of said application;

code means for providing, for each of the application-level users, respective sets of one or more application instances of each resource class component for the application on one or more machines, to service the incoming requests from the application-level users to use the application;

code means for monitoring, for each of the application-level users, the resources currently



available and resources currently consumed by the requests serviced by application instances of the resource class components of the application; and

code means for maintaining (i) a record of resources currently available to respective application-level users; and (ii) a record of resources currently consumed by respective application-level users; both records of said resources being maintained in respect of each of the one or more application instances of each resource class components[[]];

code means for adjusting the respective numbers of said one or more application instances of each resource class component; and

code means for collectively and automatically allocating fractions of different resource class components to a particular application-level user in response to a fluctuating number of application instances of one or more resource class components,

wherein said application instances of each resource class component are adjusted for each application-level user based (i) at least partly on said records of resources currently available and currently consumed by respective application-level users, and (ii) at least partly on predetermined information that estimates the number of each resource class components required to service requests for said application instances of the resource class components.